Information Assurance (IA)

Introduction Lesson Objectives

- Meet the Instructor
- Course Objectives
- Student Introductions
- Course Outline

Course Objectives

The purpose of this module is to provide students with the fundamentals of Information Assurance with regard to DOD and network security.

Topics include: Cryptography, Public Key Infrastructure (PKI), IPSec, Secure Socket Layer (SSL), Pretty Good Privacy (PGP), Firewalls, Intrusion Detection Systems (IDS), Virtual Private Networks (VPN), Defense Messaging System (DMS), the DOD accreditation process, and military COMSEC, cryptographic systems, and procedures.

Course Objectives (cont)

Evaluation is based on homework assignments, practical exercises, completion of the IASO CBT and an exam. Students will participate in two hands-on practical exercises to give them experience on the implementation, loading and usage of military cryptographic equipment.

Upon completion of this module, students will be able to select, acquire and implement the security necessary to secure a communications link. Students will also be familiar with different means to encrypt a network such as PKI, IPSec and SSL. Finally, students will have a good foundation in the DOD Information Security Certification and Accreditation Process.

Course Outline

The purpose of this module is to provide students with the fundamentals of Information Assurance with regard to DOD and network security.

Topics include: Cryptography, Public Key Infrastructure (PKI), IPSec, Secure Socket Layer (SSL), Pretty Good Privacy (PGP), Firewalls, Intrusion Detection Systems (IDS), Virtual Private Networks (VPN), Defense Messaging System (DMS), the DOD accreditation process, and military COMSEC, cryptographic systems, and procedures.

Information Assurance is . . .

Information Operations (IO) that protect and defend info and info systems by ensuring their

- confidentiality,
- authentication,
- integrity,
- availability, and
- non-repudiation.

This includes providing for restoration of info systems by incorporating

- protection,
- detection, and
- reaction capabilities.

Definition from National Information Systems Security (INFOSEC) Glossary, NSTISSI No. 4009, Aug 1997

DOD Information Assurance

Information Assurance

Defined: Based on perspective...

who has rights to access something

Defense In Depth: strategy for IA, layered approach – (think castles) built on high ground, successive barriers, water obstacles, moats, ditches, successive rings of strong walls and towers.

Today – people, operations, technology

Army IA Policy (1 of 2)

- Training
  - IASO CBT (level 1)
  - SA/NM course (level 2)
  - IAM course
  - CNDC (level 3)
  - Users (ia.gordon.army.mil)
- Mission Assurance Category
  - MAC I
  - MAC II
  - MAC III

From AR 25-2

Army IA Policy (2 of 2)

- Confidentiality Level
  - High - classified
  - Medium - sensitive
  - Basic – public or unclassified
- Robustness Level
  - High
  - Medium
  - Basic
- Level of Total System Exposure

Fundamental Tenets of IA

- Confidentiality
- Integrity
- Authentication
- Non-repudiation
  - Availability
  - Auditing
  - Access Control

Fundamentals of Cryptography

Module 1

Lesson Objectives

- Recognize adversaries and threats to information systems
- Introduce basic concepts of cryptology
- Explain how conventional encryption works
- Explain public key encryption
- Realize how hashing works and what it is used for
- Realize how key length determines security

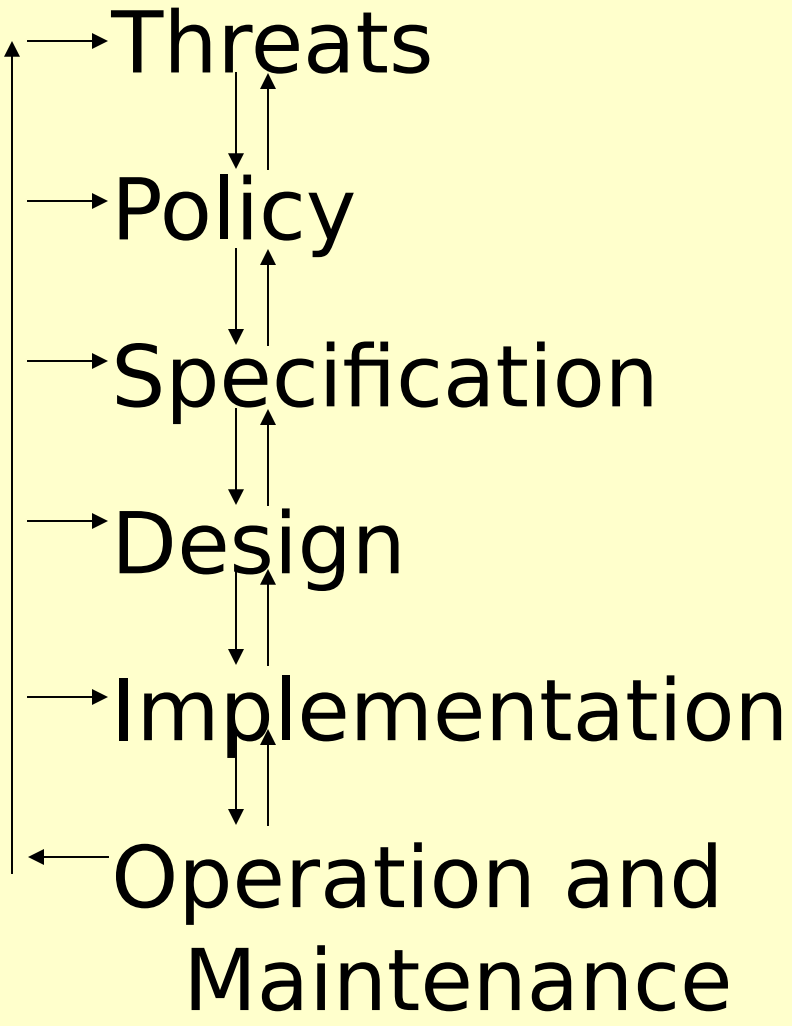
Outline

- Security Goals
- Adversaries
- Threats
- Security/Cryptography Terminology
- Cryptographic History
- Cryptographic Goals
- Summary

Security Goals

- Prevention
- Detection
- Recovery
  - Stop attack, repair damage
  - System continues to function
- Assurance

Security Life Cycle



Adversaries (1 of 2)

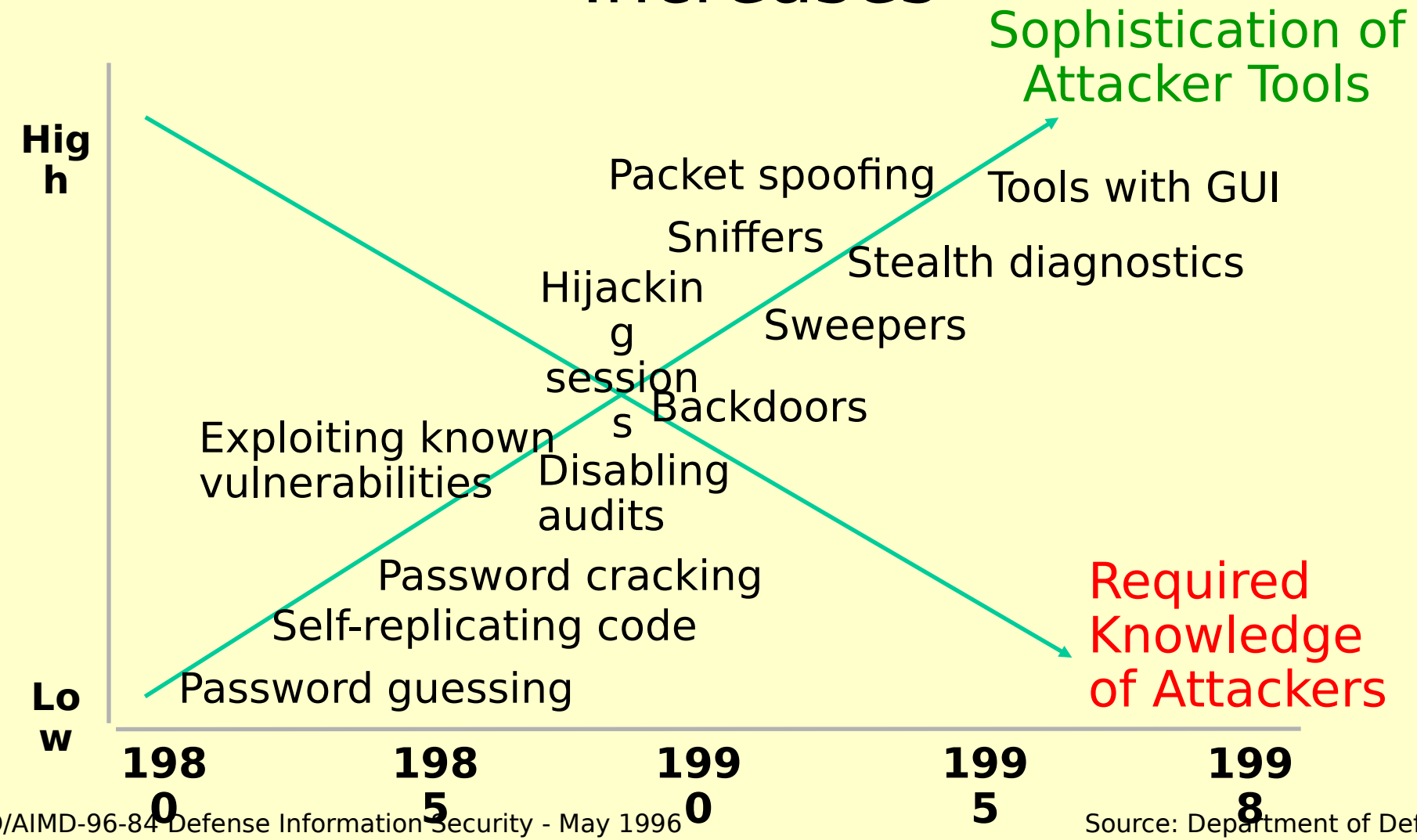
- Motivation
- Frequency
- Hackers (Tier 1, 2, 3)
- Other types:
  - Lone Criminals
  - Malicious Insiders
  - Industrial Espionage
  - Press

Adversaries (2 of 2)

Still Others…..

- Organized Crime
- Terrorists
- National Intelligence Agencies
- Infowarriors -- Nations
- Inept Users
  - Social Engineering

Attackers Require Less Knowledge as Tool Sophistication Increases



Source: Department of Defense

Threats (1 of 2)

- Natural/Environmental
  - Consequence of natural phenomena
- Poor administration
  - Default operating system installation
  - Default Microsoft IIS installation
  - Unnecessary services or daemons operating (ports)
  - Failure to keep up with security patches
  - Security policies
- Users
  - Untrained/inept
  - Malicious insider
  - Social Engineering

Threats (2 of 2)

- Software
  - Security Defective Software
    - BIND (Domain Name Server)
    - CDE ToolTalk
    - Microsoft IIS (Windows 2000 and NT 4.0)
    - Solaris Sadmin
    - UNIX r commands
  - LanMan
  - Buffer overflows

Buffer Overflow

- What is a buffer?
  - Buffer – continuous block of memory used to store a variable
  - Example – char password[8];
- What is a stack?
  - Dynamic region of memory used as a storage of temporary information.
- What is a buffer overflow?
  - Information inputted to the variable is greater that the memory allocated to the variable causing the program to overwrite other parts of the memory stack.
- Buffer overflow example

Malicious Software (Malware)

Virus – program that that makes a copy of itself in such a way to infect parts of the OS and/or applications; typically needs a host and must be executed

- Melissa
- I Love You or Love Bug

Worm – program that copies itself over computer networks, infecting programs and machines in remote locations; runs on its own

- Code Red
- Klez
- SQL Slammer
- MSBlaster
- Sobig (1)
- Bagle, MyDoom (2), NetSky (3)

Virus versus Worms

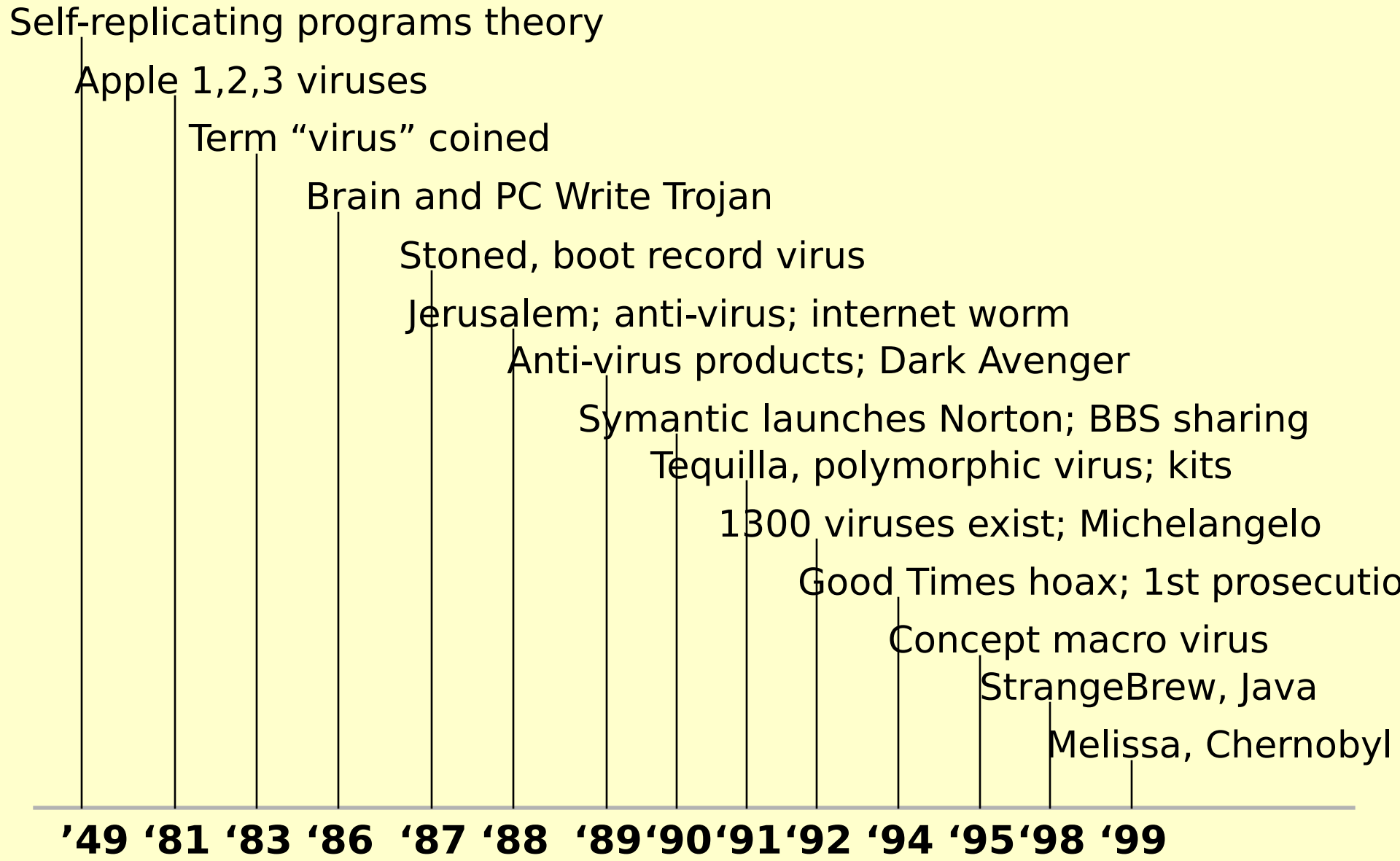
Virus -

- normally attached to executable or macro files
- normally requires user interaction to execute and spread
- primarily transport mechanisms for malicious code and may not necessarily be malicious in themselves
- Many have no payload

Worm -

- normally self-contained
- does not require user interaction to execute and spreads system to system
- Affects systems not files
- primarily transport mechanisms for malicious code and may not necessarily be malicious in themselves
- Many have no payload

Virus Timeline (1 of 2)


Self-replicating programs theory
Apple 1,2,3 viruses
Term “virus” coined
Brain and PC Write Trojan
Stoned, boot record virus
Jerusalem; anti-virus; internet worm
Anti-virus products; Dark Avenger
Symantic launches Norton; BBS sharing
Tequilla, polymorphic virus; kits
1300 viruses exist; Michelangelo
Good Times hoax; 1st prosecutio
Concept macro virus
StrangeBrew, Java
Melissa, Chernobyl
’49 ‘81 ‘83 ‘86 ‘87 ‘88 ‘89 ‘90 ‘91 ‘92 ‘94 ‘95 ‘98 ‘99

From InfoAge Business Development Service

Virus Timeline (2 of 2)

2000

- Love Bug
- Resume
- Stages
- Palm PDA “Liberty”

2001

- Anna Kournikova
- Homepage
- Code Red I and II
- Nimda
- Computer Assoc “Top 10” list

2002

- Klezworm
- Nimda
- Lioton

2003

- Ganda
- Slammer
- Sobig

2004

- Erkez
- Sasser
- Netsky
- MyDoom

Malware Types (1 of 3)

Trojan Horse – malicious, security-breaking program that is disguised as something benign

- Sub-seven
- Back orifice
- NetBus

Time Bomb – lies dormant until a specified system time or date, activating the malicious code

- Chernobyl
- Y2K

Logic Bomb – virus set for a timed release, when it "detonates," it deliberately disrupts, modifies, or erases data.

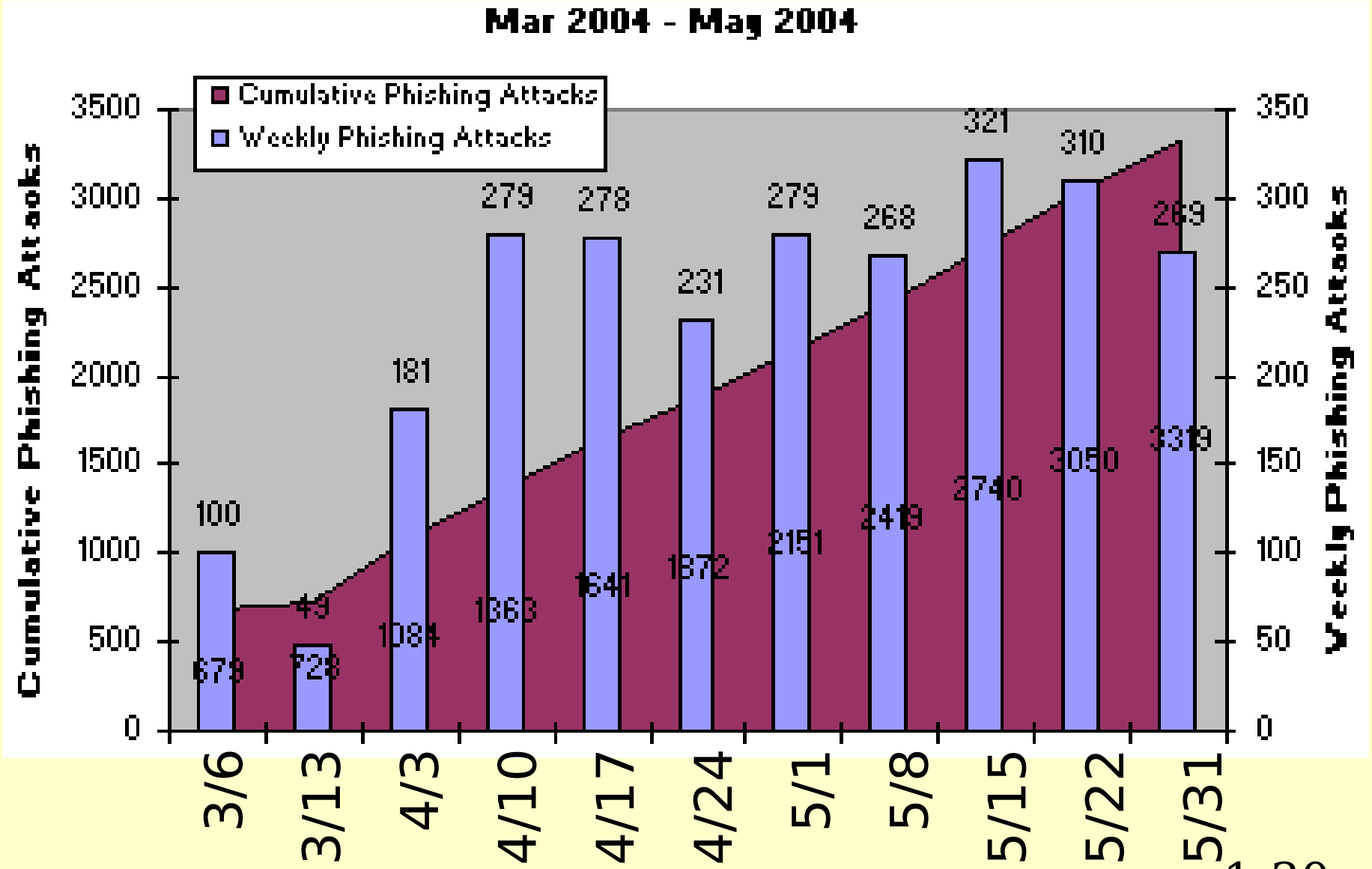
Malware Types (2 of 3)

- **Hoax** - email sent in chain letter fashion describing some devastating, highly unlikely type of virus, or propagating email forwards; noted by the Hook – Threat – Request format.
  - Bad Times
  - Outback Steakhouse or Gap email tracking
  - $200 Nieman-Marcus Cookie recipe
- **Joke -** harmless program that causes various benign activities to display on your monitor
- Check the validity before forwarding
  - www.symantec.com/avcenter/hoax.html
  - hoaxbusters.ciac.org

Malware Types (3 of 3)

- **Spyware -** Stand-alone programs that secretly monitor system activity.
- **Adware -** Programs that secretly gather personal information through the Internet and relay it back to another computer.
- **Cookie -** text files that are placed on a user's computer by Web sites that the user visits
- **Spam -** unsolicited e-mail generated to advertise some service or product
- **Phish –** spoofed emails to gather personal information from recipients

Phishing Statistics



SANS Top 10 Windows Critical Internet Security Vulnerabilities

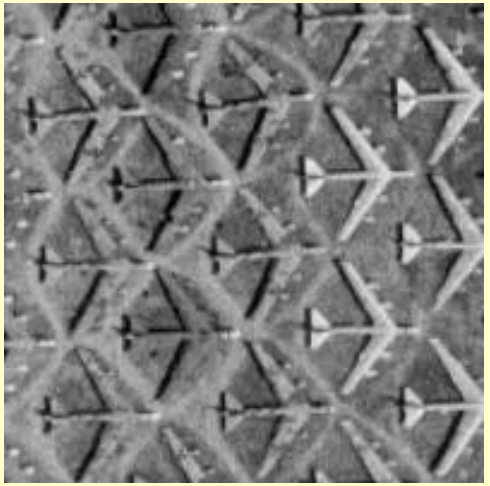
1. Internet Information Services (IIS)
2. Microsoft SQL Server (MSSQL)
3. Windows authentication
4. Internet Explorer
5. Windows Remote Access Services
6. Microsoft Data Access Components (MDAC)
7. Windows scripting host (WSH)
8. **Microsoft Outlook and Outlook Express**
9. **Windows Peer-to-peer file sharing (P2P)**
10. **Simple Network Management Protocol (SNMP)**

SANS Top 10 Unix Critical Internet Security Vulnerabilities

1. BIND Domain Name System
2. Remote Procedure Calls (RPC)
3. Apache Web Server
4. **General Unix authentication Accounts with No/Weak Passwords**
5. **Clear Text Services**
6. Sendmail vulnerabilities
7. Simple Network Management Protocol (SNMP)
8. Secure Shell
9. **Misconfiguration of Enterprise Services NIS/NFS**
10. **Open Source Sockets Layer (SSL)**

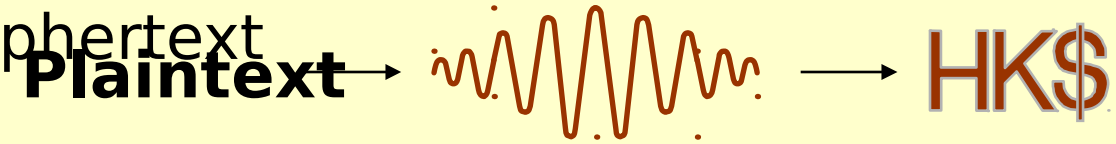
Terminology

- Threats
- Attacks
- Cryptography
- Cryptanalysis
- Steganography
- Ciphertext
- Encrypt
- Decrypt

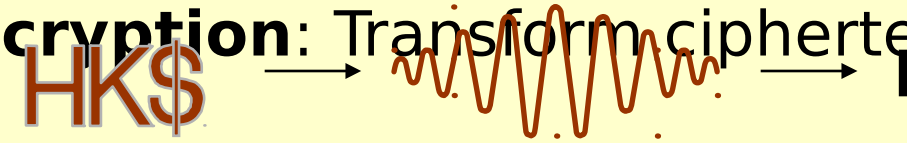


Cryptography (Secret Writing)

- **Encryption**: Transform plaintext into ciphertext



- **Decryption**: Transform ciphertext into plaintext



- **Algorithms**: Control specific encryption and decryption process

Ciphers

- **Substitution**: 1 for 1 replacement of bits or bytes
  - Caesar Cipher shifted up 3
    The enemy is nigh = Wkh hqhpb lv qljk
- **Transposition**:
  - Transposition rotate 4 characters right
    The enemy is nigh = nem yisni gh thee
- **Substitution and Transposition:** (modern algorithms)
    The enemy is nigh = hqh pblvq lj kwkh
- **Stream cipher** – one bit or byte at a time
- **Block cipher** – encrypt in chunks of data (64 or 128)

Algorithms

- A math formula that determines how data is encrypted or decrypted with a key.
- Three primary algorithm types:
  - Symmetric or conventional encryption or secret key
  - Asymmetric or public key encryption
  - Hashing
- Two ways to attack an encrypted message.
  - Systematic trial of each key used in the algorithm
  - Figure out a way to solve the algorithm without going through every calculation
- Randomness is important.

Cryptographic Techniques

- Three primary mechanisms
  - Conventional encryption (secret key or symmetric encryption)
  - Public key cryptography (asymmetric)
  - Hash function
- Multiple ways to do these
  - Several types of algorithms are used for each type of encryption.

Symmetric Encryption (1 of 2)

- Encryption and decryption use the same key
- Provides confidentiality and basic authentication
- Does not provide non-repudiation
- Normally uses a proven algorithm to encrypt or decrypt

Symmetric Encryption (2 of 2)

- **Advantages**
  - Faster – smaller key lengths.
  - Math algorithm is straight forward; usually cannot be broken
- **Disadvantages**
  - No sender authentication; no digital signature capability
  - Breaking one key compromises all users
  - Does not scale well

Asymmetric Encryption

- **Digital Certificate** – normally a set of keys, 1 private, 1 public (with other information).
- **Public key encryption is asymmetric**
  - Does not use the same key to encrypt/decrypt.
  - Ensured by the math algorithm; prime numbers
  - Data encrypted with a public key cannot be decrypted with the same public key.
  - Allows public keys to be shared.
  - Public key of the recipient is used to encrypt

Asymmetric Encryption with Authentication

Public Keys can be used to authenticate messages.

- Data signed with a private key of the sender, can only be verified with the sender's public key.
- Private key of the sender is not used to encrypt but to digitally sign.
- Anyone can verify a private key.

With Asymmetric Encryption

- Encryption uses the public key of the recipient.
- Decryption uses the private key of the recipient.
- Signing uses the private key of the sender.
- Authentication uses the public key of the sender.

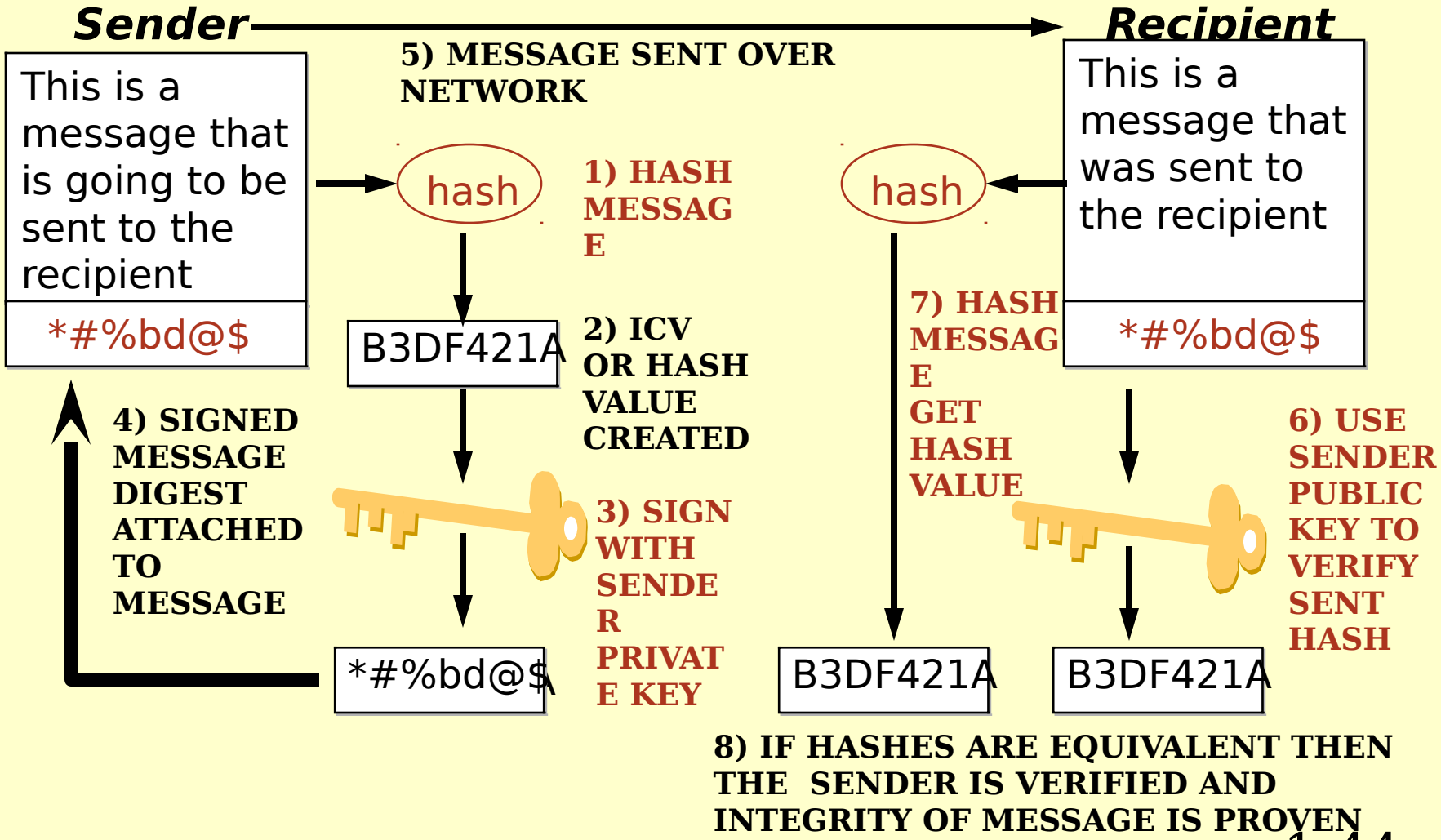
Hash Function

- Sender and recipient use the same key to compute a hash value (or ICV) or a message digest (using SHA-1 or MD-5).
- Key known only to the sender and recipient.
- ICV is computed on the message to be sent (by the sender) and attached to the message.
- If message is changed by just 1 bit the ICV is altered.
- If ICV computed by recipient matches ICV received, then message is known to be unchanged in transmission.

Digital Signatures

- Verifies the sender of a message.
- A specific bit patter is attached to a message, resulting in an ICV or MD.
- This MD, not the message its appended to, is signed with a sender’s private key.
- Receiver takes the MD and verifies the sender by using their public key.
- If MD matches the message has not been altered and the sender is authenticated.
- Provides the foundation of non-repudiation.

Digital Signatures and Integrity


Sender
Recipient
This is a message that is going to be sent to the recipient
*#%bd@$
5) MESSAGE SENT OVER NETWORK
hash
1) HASH MESSAGE
B3DF421A
2) ICV OR HASH VALUE CREATED
3) SIGN WITH SENDER PRIVATE KEY
*#%bd@$
4) SIGNED MESSAGE DIGEST ATTACHED TO MESSAGE
hash
7) HASH MESSAGE GET HASH VALUE
B3DF421A
This is a message that was sent to the recipient
*#%bd@$
6) USE SENDER PUBLIC KEY TO VERIFY SENT HASH
B3DF421A
8) IF HASHES ARE EQUIVALENT THEN THE SENDER IS VERIFIED AND INTEGRITY OF MESSAGE IS PROVEN

Key Length (1 of 2)

- Difference in key length is significant
  - 40 bits = 2^40 = 1,099,511,627,776
  - 128 bits = 2^128 = 34,028,200,000,000,000,000,000,000,000,000,000,000
- 128 bit keys are therefore
  - 309,000,000,000,000,000,000,000,000 times harder to crack than a 40 bit key.
- Comparison if a key can be broken in an algorithm:
  - Using 40 bits in 5 seconds
  - Then 128 bits will take 49,068,526,417,640,960,921 years
- Difference between weak and strong is BIG

Key Length (2 of 2)

- Not all algorithms can use every number in the key length so it does vary by encryption algorithm and method of encryption
- Key Space
- Equivalent strength between public and conventional keys.

Conventional Key	Public Key	Person	Military
40	274	days	microseconds
56	384	years	seconds
64	512	millennia	minutes
80	1024	infeasible	centuries
112	2048	infeasible	infeasible
128	3072	infeasible	infeasible

Public Key Infrastructure (Why the I?)

The I in PKI is critical.

- Provides
  - issue/update keys
  - repository for public keys
  - information on revoked keys
- Without an easy way to obtain public keys and verify currency, public key encryption is vulnerable and not likely to be used

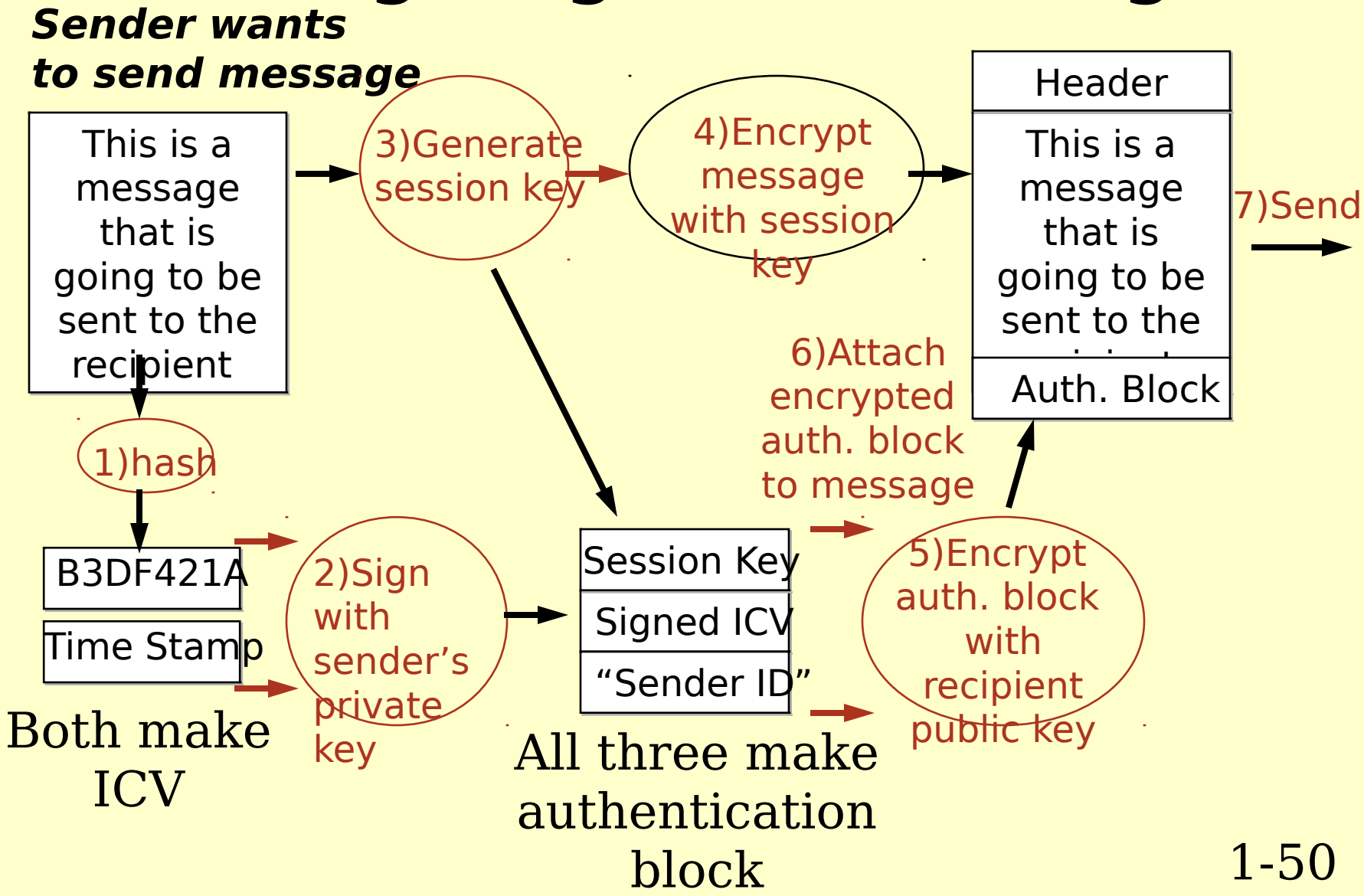
Public Key Algorithms – Trade Offs

- Public key algorithms are slower than symmetric encryption algorithms due to longer bit lengths.
- Public key algorithms however provide the following services:
  - Confidentiality
  - Authentication
  - Non-repudiation
  - Ease of use with revocation lists

Public Key Algorithms Signing and Sealing

- Due to speed, most messages using public keys are encrypted with a conventional encryption key.
- An authentication block is attached to the message, this authentication block includes:
  - The conventional encryption (session) key.
  - The message is already signed with the sender’s private key.
  - A way to identify the sender.
- The authentication block is encrypted with the receiver’s public key.

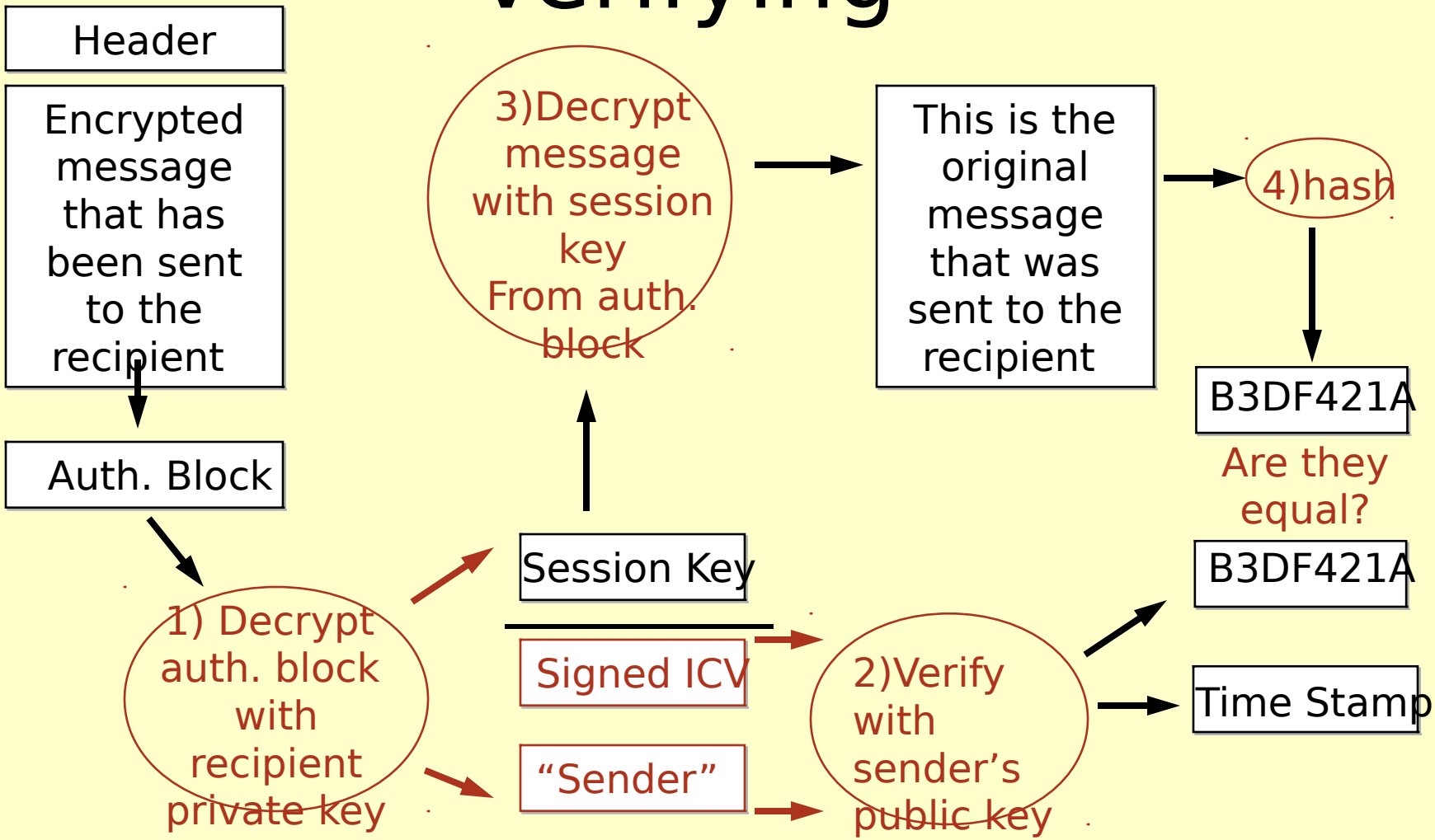
PK Signing and Sealing


Sender wants to send message
This is a message that is going to be sent to the recipient
1)hash
B3DF421A
Time Stamp
Both make ICV
2)Sign with sender's private key
3)Generate session key
4)Encrypt message with session key
Session Key
Signed ICV
"Sender ID"
All three make authentication block
5)Encrypt auth. block with recipient public key
6)Attach encrypted auth. block to message
Header
This is a message that is going to be sent to the
Auth. Block
7)Send

Public Key Algorithms Unsealing

- The authentication block is detached and decrypted with recipient's private key. The recipient now has:
  - A signed hash value.
  - The name of the sender.
  - The conventional (session) key used to encrypt the message.
- The session key is then used to decrypt main message.
- The recipient then uses public key of sender to authenticate sender and verify integrity of message.
- Two examples - PGP uses IDEA for conventional encryption and RSA normally uses DES.

PK Unsealing and Verifying



Security Services Supported by Cryptography

- Confidentiality – Encryption whether with a public or conventional algorithm.
- Integrity – Hash, message digest or ICV.
- Authentication – Verifying the sender by an assumption or by a signature.
- Non-repudiation – Validating a digital signature where no one can disclaim it.